Orion Threat Alert: Flight of the BumbleBee

Cynet.com/orion-threat-alert-flight-of-the-bumblebee/

April 14, 2022

Orion Threat Alert



By: Max Malyutin – Orion Threat Research Team Leader

Orion, Cynet's Threat Research and Intelligence team, spotted a new malware campaign in the wild: BumbleBee.

Wondering what's going on? Let us fill you in.

We noticed a new trend in Initial Access Brokers' (IAB) tactics to gain access to their victims' machines. Initial Access Brokers refers to a cybercrime group that specializes in gaining initial access to organizations for the sole purpose of offering it to other threat actor groups. The trend started earlier this year and our team recently spotted their new BumbleBee campaign.

Usually, we observe malicious spam (MalSpam) campaigns that deliver malicious documents (MalDoc) to lure the victims to interact with the MalDoc and execute the malicious macro code by clicking "Enable Content." That in turn downloads and executes the malicious payload, for example, <u>the notorious Emotet campaigns</u>.

We expected these groups to change the initial access methods. We believe there is a direct relation to the changes Microsoft applied recently to the default policy in their Office products: "<u>Macros from the internet will be blocked by default in Office</u>" and "<u>Excel 4.0 (XLM)</u>

macros are disabled by default." These changes impact IABs because they have been abusing Office documents with malicious macros for years.

It appears that they've come up with a plan B.

In this post, we will cover what this campaign is, and how the IAB distributes the BumbleBee malware and its TTPs. We will also explain each TTP according to the MITRE ATT&CK model, and its purpose.

A new campaign in the wild: BumbleBee

From our initial analysis, BumbleBee is a custom new loader that is used by different IAB groups. This malware was observed injecting <u>Cobalt Strike</u> shellcodes in memory and using several tactics, techniques, and procedures (TTPs) in order to compromise the victim's environment.

As part of the campaign, the threat actors abuse spoofed companies' identities (like fake employee email addresses, fake websites, etc.) and use legitimate public storage services to deliver the malicious ISO image file. The ISO image file is responsible for luring the victim to execute the BumbleBee malware.

We've seen Living Off the Land Binaries (<u>LOLBins</u>) execution with <u>rundll32</u>, which allows threat actors to avoid defenses. BumbleBee also creates a <u>scheduled task</u> on the compromised host for persistence and executes a <u>Visual Basic script</u> via the scheduled task. The IAB relies on the <u>user (victim) execution to execute</u> the BumbleBee payload by luring the victim to mount an ISO image file and click on a Windows shortcut (LNK) file.

The malware name, BumbleBee, was chosen because of its unique user agent, "bumblebee," that was used as part of the communication with the command and control server (C2).

Threat Analysis Group (TAG) shared observations on the financially motivated threat actor, <u>EXOTIC LILY</u>, that use the BumbleBee malware. In addition, TAG mentioned an interesting point of collaboration between EXOTIC LILY and the <u>WIZARD SPIDER</u> threat group.

Orion's observations

This type of attack is new, and the cybersecurity community is still gathering data to glean more insights on the nature of this attack and its targets.

Orion found a high number of targeted companies based in the US with the following distribution method that delivers the BumbleBee malware: Spear phishing email > URL Link (TransferXL, TransferNow, WeTransfer) > Zipped ISO > ISO (contains the LNK file and the BumbleBee payload).

You can see the execution flow in the image below.



The infection flow

We've handled several incident response (IR) cases where threat actors distributed BumbleBee malware. After the initial infection, the threat actors inject Cobalt Strike shellcode in memory and execute discovery commands to collect info about the victim's network. We believe that threat actors performed this data collection in order to execute the next stage of the infection.

The next stage is probably related to ransomware operations. We're still investigating IR cases in order to find conclusive evidence that the next stage delivers ransomware.

On April 12, 2022, the BumbleBee IAB group was spotted using IMG file format in addition to ISO file format.

⊚ File	Image Image File Home Share View Drive Tools							
\leftarrow	\rightarrow \land \uparrow \bigcirc \rightarrow This	PC > DVD Drive (E:)						
>	Documents ^	Name	Date modified	Туре	Size			
>	🕂 Downloads	🗟 autorun.dll	4/7/2022 6:35 AM	Application exten	2,867 KB			
>	🜟 Favorites	📕 documents	4/7/2022 6:35 AM	Shortcut	2 KB			
	🔁 Links							
>	📕 Local Settings							
>	MicrosoftEdge							
>	My Document:							
>	NetHood							
	OneDrive							
>	Pictures							

You can see an example in the image below.

The IMG file, which contains LNK and DLL

Initial Access

The BumbleBee payload was delivered via a spear phishing email that was sent from a spoofed email address. The email contains a URL link to the legitimate public storage service, TransferXL.



Spear phishing email with a link to TransferXL

Below you'll see the legitimate public storage site, which leads the victim to the link to the malicious file.



TransferXL legitimate public storage services

Once they click download, the victim receives a ZIP folder that contains the malicious ISO image files.



Spoofed company email address

Execution

Below is an example of what the ZIP file from the TransferXL link looks like.



ZIP file download from TransferXL

The ZIP file contains an ISO image file with the following name "*documents-04-106.iso.*" Note that the following ISO image file name pattern was used for all the files that we have analyzed:

documents-[0-9]{1,4}-[0-9]{1,4}\.iso

TransferXL-00jLBF6Rtw8vrX.zip (evaluation copy) File Commands Tools Favorites Options Help a:3 ~ ----Ī E (1) Test View Find Wizard VirusScan Comment SFX Add Extract To Delete Info ↑ TransferXL-00jLBF6Rtw8vrX.zip - ZIP archive, unpacked size 2,910,208 bytes Modified Downloaded using TransferXL Name Size Packed Type CRC32 .. 1,879,389 Disc Image File documents-04-106.iso 2,910,208 E08AE662

ISO image file

From this step, threat actors rely on the victim (user) interaction with the ISO image file. The threat actors use a masquerading technique by setting the LNK file icon to be a folder icon in order to lure the victim to click on the LNK file:



ISO image file contains LNK and DLL

In addition, the DLL payload attribute is set as "Hidden" in order to hide the DLL payload from the user when interacting with the ISO image file:

15	settings.dll	Prop	perties

General Detai	ls				
Sile I	settings.dll				
Type of file:	Application extension (.dll)				
Opens with:	JetBrains dotPeek Change				
Location:	E:\				
Size:	2.72 MB (2,856,448 bytes)				
Size on disk:	2.72 MB (2,856,960 bytes)				
Created:	Wednesday, April 6, 2022, 4:20:58 AM				
Modified:	Wednesday, April 6, 2022, 4:20:58 AM				
Accessed:					
Attributes:	Read-only Hidden Archive				
	OK Cancel Appl				

 \times

Hidden attribute for the DLL

The masqueraded LNK file properties show that the execution target is as follows:

C:\Windows\System32\rundll32.exe settings.dll,IternalJob

documents P	roperties	>
General Shortcu	ut Details	
do	cuments	
Target type:	Application	
Target location:	System32	
Target:	s\System32\run¦lll32.exe settings.dll,IternalJob]
Start in:]
Shortcut key:	None]
Run:	Normal window	
Comment:]
Open File Lo	cation Change Icon Advanced	I
	OK Cancel Apply	

LNK executes the DLL via rundll32 command

After the initial execution, the BumbleBee DLL is copied to the %programdata%/{RandomDir} directory. In addition to the DLL, a VBS script is also dropped to the same directory:

 $\label{eq:a-z}:\programdata\[a-z0-9]{16}\[a-z0-9]{16}\]$

action	process_path	file_path	new_process_command_line
Write	c:\windows\system32\rundll32.exe	c:\programdata\ <mark>045241ad770d73bd</mark> \8cbc96f0e4e2ae45.vbs	-
Create New	c:\windows\system32\rundll32.exe	c:\programdata\ <mark>045241ad770d73bd</mark> \8cbc96f0e4e2ae45.vbs	-
Execute New Process	c:\windows\system32\wbem\wmiprvse.exe	c:\windows\system32\wscript.exe	wscript.exe C:\\ProgramData\\ <mark>045241ad770d73bd</mark> \\8cbc96f0e4e2ae45.vbs
Execute New Process	c:\windows\explorer.exe	c:\windows\system32\rundl132.exe	'C:\\Windows\\System32\\rundll32.exe' settings.dll, <mark>IternalJob</mark>
Execute New Process	c:\windows\explorer.exe	c:\windows\system32\rundl132.exe	'C:\\Windows\\System32\\rundll32.exe' settings.dll, <mark>IternalJob</mark>
Execute New Process	c:\windows\explorer.exe	c:\windows\system32\rundll32.exe	'C:\\Windows\\System32\\rundll32.exe' settings.dll, <mark>IternalJob</mark>

TTPs indicators during the execution

We have other artifacts from different IR cases, where we have observed the following activity. The screenshot below shows an event that detected a creation of a payload in the %ProgramData%\{Random} directory the DLL payload is a copy of the initial BumbleBee

loader that executed by Rundll32 from the ISO image file:



Copy of the BumbleBee DLL to %Programdata% directory

In other IR cases, we observed an execution flow that's bit different. For example, a LNK that points to the following execution targets:

- cmd.exe /c start rundll32 neqw.dll,IternalJob
- rundll32.exe advpack.dll,RegisterOCX sysctl.exe

Persistence

We detected a scheduled task execution during the BumbleBee infection:

Grandparent process: svchost.exe -k netsvcs -p -s Schedule

Parent process:

wscript.exe [a-z]:\\programdata\\[a-z0-9]{16}\\[a-z0-9]{16}\.vbs

Child process:

rundll32.exe [a-z]:\\programdata\\[a-z0-9]{16}\\[a-z0-9]{16}\.dll,{Export}

's'	.rdata:00000001801D0010	A0000000	С	ntdll.dll
's'	.rdata:00000001801D0020	0000005	С	.dl
's'	.rdata:00000001801D0028	0000005	С	.vbs
's'	.rdata:00000001801D0030	000006C	С	$Set \ objShell = CreateObject(\"Wscript.Shell")\r\nobjShell.Run \"rundll32.exe \ my_application_path, \ IternalJob\"\r\noble application_path, \ IternalJob\"\$
's'	.rdata:00000001801D00A0	00000014	С	my_application_path
's'	.rdata:00000001801D00B8	00000018	C (1	wscript.exe
's'	.rdata:00000001801D00D0	000000D	С	wscript.exe

Strings from the BumbleBee loader show the VBS script and the execution method

We also observed WMI execution. The VBS file that was executed via a scheduled task, was also executed through WMI:

Grandparent process:

svchost.exe -k DcomLaunch

Parent process:

wmiprvse.exe -Embedding

Child process:

```
wscript.exe [a-z]:\\programdata\\[a-z0-9]{16}\\[a-z0-9]{16}\.vbs
```

's'	.rdata:00000001801D70E0	00000016	C (1	ROOT\\CIMV2
's'	.rdata:00000001801D70F8	00000014	C (1	ole32.dll
's'	.rdata:00000001801D7110	00000012	С	CoSetProxyBlanket
's'	.rdata:00000001801D7128	000000E	C (1	Create
's'	.rdata:00000001801D7138	0000001C	C (1	Win32_Process
's'	.rdata:00000001801D7158	0000002A	C (1	Win32_ProcessStartup
's'	.rdata:00000001801D7184	0000004	C (1	

Strings from the Bumblebee loader show the WMI Win32_Process execution

Defense Evasion

In our labs, we observed that BumbleBee uses several anti-VM methods to avoid detection.

One of the anti-VM checks is related to the VirtualBox product:



Check for IpWindowName if matches VirtualBox

Other anti-VM artifacts were found after unpacking, as can be seen in the following strings:

Offset	Туре	Strings found
001D8573	UNICODE	VBOX_
001D85AB	UNICODE	VBOX
001D85E3	UNICODE	VBOX
001D8BE9	UNICODE	VBoxControl.exe
001D868E	UNICODE	VBoxGuest
001D8CDC	UNICODE	VBoxGuest
001DA478	UNICODE	VBoxGuest
001D8891	UNICODE	VBoxGuest.sys
001D8D01	UNICODE	VBoxMiniRdDN
001D8CB4	UNICODE	VBoxMiniRdrDN
001D86DE	UNICODE	VBoxMouse
001DA490	UNICODE	VBoxMouse
001D8851	UNICODE	VBoxMouse.sys
001D878E	UNICODE	VBoxSF
001DA468	UNICODE	VBoxSF
001D88D1	UNICODE	VBoxSF.sys
001D872E	UNICODE	VBoxService
001D8D2C	UNICODE	VBoxTrayIPC
001D8D51	UNICODE	VBox TrayIPC
001D8DD0	UNICODE	VBox Tray Tool Wnd
001D8DA0	UNICODE	VBox Tray Tool Wnd Class
001D87DE	UNICODE	VBoxVideo
001D8909	UNICODE	VBoxVideo.sys
001D8F58	UNICODE	VBoxVideoW8
001D8F70	UNICODE	VBoxWddm
001D9A68	UNICODE	VMSrvc.exe
001D9A80	UNICODE	VMUSrvc.exe
001D9A60	ASCII	VMWARE
001D92B8	UNICODE	VMWARE
001D9938	UNICODE	VMWare
001D9868	UNICODE	VMWare\
001D9A58	ASCII	VMware
001D94F9	UNICODE	VMware, Inc.\VMware Tools
001D9948	UNICODE	\\.\HGFS
001D8CD8	UNICODE	\\.\VBoxGuest

List of strings that are related to VMware and VirtualBox

BumbleBee also detects if it is running within a VM by checking for known services that are related to different VM products:

Offset	Туре	Strings recognized as registry key	
001D9AF0	UNICODE	SOFTWARE\Microsoft\Virtual Machine\Guest\Parameters	
001D9410	UNICODE	SYSTEM\ControlSet001\Control\SystemInformation	
001D9E20	UNICODE	SYSTEM\ControlSet001\Services\BALLOON	
001D9E70	UNICODE	SYSTEM\ControlSet001\Services\BalloonService	
001D8670	UNICODE	SYSTEM\ControlSet001\Services\VBoxGuest	
001D86C0	UNICODE	SYSTEM\ControlSet001\Services\VBoxMouse	
001D8770	UNICODE	SYSTEM\ControlSet001\Services\VBoxSF	
001D8710	UNICODE	SYSTEM\ControlSet001\Services\VBoxService	
001D87C0	UNICODE	SYSTEM\ControlSet001\Services\VBoxVideo	
001D9D60	UNICODE	SYSTEM\ControlSet001\Services\VirtIO-FS Service	
001D9DC0	UNICODE	SYSTEM\ControlSet001\Services\VirtioSerial	
001D9ED0	UNICODE	SYSTEM\ControlSet001\Services\netkvm	
001D9CC0	UNICODE	SYSTEM\ControlSet001\Services\vioscsi	
001D9D10	UNICODE	SYSTEM\ControlSet001\Services\viostor	

List of services that are related to VM products

BumbleBee checks whether certain user names reside in the victim's machine by comparing against a hardcoded list of user names. This allows BumbleBee to detect sandboxes and labs that are used for malware analysis:

's'	.rdata:00000001801D88C8	00000018	C (1	CurrentUser
's'	.rdata:00000001801D88E0	00000010	C (1	Sandbox
's'	.rdata:00000001801D88F0	000000C	C (1	Emily
's'	.rdata:00000001801D8900	00000010	C (1	HAPUBWS
's'	.rdata:00000001801D8910	00000012	C (1	Hong Lee
's'	.rdata:00000001801D8928	00000012	C (1	IT-ADMIN
's'	.rdata:00000001801D8940	00000010	C (1	Johnson
's'	.rdata:00000001801D8950	000000E	C (1	Miller
's'	.rdata:00000001801D8960	000000E	C (1	milozs
's'	.rdata:00000001801D8970	0000001A	C (1	Peter Wilson
's'	.rdata:00000001801D8990	000000C	C (1	timmy
's'	.rdata:00000001801D89A0	00000012	C (1	sand box
's'	.rdata:00000001801D89B8	00000010	C (1	malware
's'	.rdata:00000001801D89C8	00000010	C (1	maltest
's'	.rdata:00000001801D89D8	00000014	C (1	test user
's'	.rdata:00000001801D89F0	000000C	C (1	virus
's'	.rdata:00000001801D8A00	00000012	C (1	John Doe
's'	.rdata:00000001801D8A20	00000046	C (1	Checking if username matches : %s
's'	.rdata:00000001801D8A68	000000E	C (1	VMWare

List of hardcoded usernames which are related to sandboxes and labs

In addition, it uses WMI queries to collect system details and information:

- SELECT * FROM Win32_BaseBoard
- SELECT * FROM Win32_Bus

- SELECT * FROM Win32_ComputerSystem
- SELECT * FROM Win32_Fan
- SELECT * FROM Win32_NTEventlogFile
- SELECT * FROM Win32_OperatingSystem
- SELECT * FROM Win32_PnPDevice
- SELECT * FROM Win32_PnPEntity

Discovery

We found that the threat actors used the AdFind tool to enumerate and map the victim's network. The ADFind tool was found in the %ProgramData% directory.

In the instance we observed, the following commands were used:

- adfind.exe -gcb -sc trustdmp
- adfind.exe -f "(objectcategory=group)"
- adfind.exe -f "(objectcategory=organizationalUnit)"
- adfind.exe -f "objectcategory=computer"
- adfind.exe -f "(objectcategory=person)"

Command and Control

After the initial execution, the BumbleBee process (Rundll32) communicated with the Command-and-Control server (C2). We've seen several C2 servers from different IR cases:

- IP: 23.82.19[.]208:443
- IP: 192.236.198[.]63:433
- IP: 45.147.229[.]177:433

III rundll32.exe (2864) (0x14858345000 - 0x14858372000)

00000ba0 50 6d 98 1	1b fb 7f 00 00 00 00 00	00 00 00 00 00 d0 f1 34 58 4	8 01 00 00 00 00 00 00 00 00 00 00 P	2m4XH
00000bc0 40 00 00 0	00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00 00 00 00]
00000be0 00 00 00 0	00 00 00 00 00 00 00 00	00 00 00 00 00 43 00 3a 00 5	c 00 50 00 f0 95 ad b6 68 0a 00 80 .	C.:.\.Ph
00000c00 50 6d 98 1	1b fb 7f 00 00 00 00 00	00 00 00 00 00 50 f9 34 58 4	8 01 00 00 00 00 00 00 00 00 00 00 P	mP.4XH
00000c20 40 00 00 0	00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00 00 00 0	
00000c40 00 00 00 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 7	8 00 73 00 8a 95 ab b6 72 0b 00 90 .	x.sr
00000c60 43 00 3a 0	00 5c 00 57 00 69 00 6e	00 64 00 6f 00 77 00 73 00 5	c 00 73 00 79 00 73 00 74 00 65 00 C	.:.\.W.i.n.d.o.w.s.\.s.y.s.t.e.
00000c80 6d 00 33 0	00 32 00 5c 00 77 00 62	00 65 00 6d 00 5c 00 66 00 6	1 00 73 00 74 00 70 00 72 00 6f 00 m	1.3.2.\.w.b.e.m.\.f.a.s.t.p.r.o.
00000ca0 78 00 2e 0	00 64 00 6c 00 6c 00 00 0	00 6c 00 00 00 64 00 6f 00 7	7 00 73 00 84 95 al b6 79 0c 00 90 x	(d.l.lld.o.w.sy
00000cc0 40 ff 2f 5	58 48 01 00 00 40 ff 2f	58 48 01 00 00 00 00 00 00 0	0 00 00 00 01 00 00 00 00 00 00 00 0	./XH@./XH
00000ce0 04 00 00 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 90 37 34 58 4	8 01 00 00 09 00 00 00 00 00 00 00 .	74XH
00000d00 00 00 00 0	00 00 00 00 00 04 00 00	0 00 00 00 00 00 00 00 00 00 0	0 00 00 00 9e 95 bf b6 6c 0d 00 88 .	
00000d20 le c0 3e 4	42 35 2e d2 11 b6 04 00 1	10 4b 70 3e fd 00 00 00 00 3	a 20 54 68 04 5d 88 8a eb 1c c9 11 .	>B5Kp>: Th.]
00000d40 9f e8 08 0	00 2b 10 48 60 02 00 00	0 00 00 00 00 00 00 00 00 00 0	0 00 00 00 02 00 00 00 69 65 64 2e .	+.H`ied.
00000d60 00 00 00 0	00 00 00 00 00 06 00 00 0	00 01 00 00 00 01 00 00 00 6	7 00 72 00 98 95 b5 b6 44 0e 00 90 .	g.rD
00000d80 e8 b6 ba 4	4c fb 7f 00 00 c0 10 32	58 48 01 00 00 c0 10 32 58 4	8 01 00 00 60 10 32 58 48 01 00 00 .	L2XH2XH`.2XH
00000da0 01 00 00 0	00 44 4c 4c 00 03 00 00	00 72 00 6f 00 f0 df 33 58 4	8 01 00 00 60 72 32 58 48 01 00 00 .	DLLr.o3XH`r2XH
00000dc0 8c ad 1c 1	1b ab 2d d2 11 b6 04 00 1	LO 4b 70 3e fd 44 00 4b 00 5	c 00 6f 00 92 95 b3 b6 6e 0f 00 80 .	
00000de0 50 6d 98 1	1b fb 7f 00 00 00 00 00	0 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 P	?m
00000e00 04 00 00 0	00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 .	
00000e20 00 00 00 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 40 78 bf 4c 0	0 00 00 00 ac 95 89 b6 3a 10 00 80 .	@x.L
00000e40 00 5f 34 5	58 48 01 00 00 a0 64 34	58 48 01 00 00 c8 ef a7 70 3	0 00 00 00 b0 64 34 58 48 01 00 00 .	_4XHd4XHp0d4XH
00000e60 10 5f 34 5	58 48 01 00 00 01 00 00	<u>00 00 00 </u> 00 00 0c 00 09 00 4	1 01 00 00 55 73 65 72 2d 41 67 65 .	_4XHAQuser-Age
00000e80 6e 74 3a 2	20 62 75 6d 62 6c 65 62	65 65 0d <mark>0a 00 00 00 00 00 0</mark>	0 00 00 00 a6 95 87 b6 66 11 00 92 <mark>n</mark>	t: bumblebeef
00000ea0 43 00 3a 0	00 5c 00 57 00 69 00 6e	00 64 00 6f 00 77 00 73 00 5	c 00 73 00 79 00 73 00 74 00 65 00 C	.:.\.W.i.n.d.o.w.s.\.s.y.s.t.e.
00000ec0 6d 00 33 0	00 32 00 5c 00 77 00 62	00 65 00 6d 00 5c 00 77 00 6	2 00 65 00 6d 00 73 00 76 00 63 00 m	.3.2.\.w.b.e.m.\.w.b.e.m.s.v.c.
00000ee0 2e 00 64 0	00 6c 00 6c 00 00 00 6c	00 00 00 20 00 4e 00 65 00 7	4 00 77 00 a0 95 9d b6 6b 12 00 80 .	.d.l.llN.e.t.wk
00000f00 e8 ef a7 7	70 30 00 00 00 40 5e 34 :	58 48 01 00 00 50 5e 34 58 4	8 01 00 00 00 00 00 00 00 00 00 00 .	p0@^4XHP^4XH
00000f20 00 00 00 0	00 00 00 00 00 00 00 00 00	$00 \ 00 \ 00 \ 00 \ 00 \ 10 \ 00 \ 03 \ 00 \ 4$	6 00 00 00 43 6f 6e 74 65 6e 74 2d .	FContent-
00000f40 4c 65 6e 6	67 74 68 3a 20 31 37 38	0d 0a 00 00 00 56 00 42 00 5	3 00 3b 00 ba 95 9b b6 42 13 00 80 L	ength: 178V.B.S.;B
00000f60 50 6d 98 1	1b fb 7f 00 00 00 00 00	00 00 00 00 00 50 f9 34 58 4	8 01 00 00 00 00 00 00 00 00 00 00 P	?mP.4XH
00000f80 40 00 00 0	00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 00 0	<u>}</u>
00000fa0 00 00 00 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 53 00 4f 00 5	2 00 5f 00 b4 95 91 b6 43 14 00 80 .	S.C.RC
00000fc0 50 6d 98 1	1b fb 7f 00 00 00 00 00 0	00 00 00 00 00 b0 f3 34 58 4	8 01 00 00 00 00 00 00 00 00 00 00 P	?m4XH
00000fe0 40 00 00 0	00 00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 00 0	J
00001000 00 00 00 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 6c 00 36 00 3	4 00 20 00 4e 96 6f b6 6d 15 00 80 .	l.6.4N.o.m
00001020 50 6d 98 1	1b fb 7f 00 00 00 00 00	0 00 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 P	?m
00001040 04 00 00 0	00 00 00 00 00 00 00 00	0 00 00 00 00 00 00 00 00 0	0 00 00 00 00 00 00 00 00 00 00 00 .	
00001060 00 00 00 0	~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~	10 00 00 00 00 40 78 bf 4c 4	3 00 45 00 48 GE E5 NE 4F 1E 00 GO	RA TO R H = O

Example of the unique User-Agent: BumbleBee in the payload's memory

's'	.rdata:00000001801D3A88	0000006	С	/gate
's'	.rdata:00000001801D3A90	0000000A	С	bumblebee
's'	.rdata:00000001801D3AA0	000000A	С	handshake

Additional reference to the BumbleBee malware name

All the collected system and network information is sent to the C2 server, which sends back a response containing the next step/command to execute based on that info.

BumbleBee binary analysis

In this section, we will cover some interesting indicators and artifacts that highlighted the BumbleBee actions and heuristics. These artifacts also help us to identify the BumbleBee malware.

We analyzed several payloads and all of them had the same artifacts.

After unpacking the BumbleBee loader and by searching in the metadata of the unpacked payload, we identified BumbleBee's internal name, "LdrAddx64.dll," and two export functions – "IternalJob" and "SetPath."

OffsetNameValu204E20Characteristics0204E24TimeDateStamp6240204E28MajorVersion0204E2AMinorVersion0204E2CName204E204E30Base1204E34NumberOfFunc2204E38NumberOfNames2	ue Me IC968C Tue	eaning esday, 05.04.2022	19:20:44 UTC	
204E20Characteristics0204E24TimeDateStamp6240204E28MajorVersion0204E2AMinorVersion0204E2CName204E204E30Base1204E34NumberOfFunc2204E38NumberOfNames2	IC968C Tue	esday, 05.04.2022	19:20:44 UTC	
204E24TimeDateStamp6240204E28MajorVersion0204E2AMinorVersion0204E2CName204E204E30Base1204E34NumberOfFunc2204E38NumberOfNames2	C968C Tue	esday, 05.04.2022	19:20:44 UTC	
204E28 MajorVersion 0 204E2A MinorVersion 0 204E2C Name 204E 204E30 Base 1 204E34 NumberOfFunc 2 204E38 NumberOfNames 2				
204E2A MinorVersion 0 204E2C Name 204E 204E30 Base 1 204E34 NumberOfFunc 2 204E38 NumberOfNames 2	IFFC 1.4			
204E2C Name 204E 204E30 Base 1 204E34 NumberOfFunc 2 204E38 NumberOfNames 2	ESC 1.4.			
204E30 Base 1 204E34 NumberOfFunc 2 204E38 NumberOfNames 2	EDC Lar	rAddx64.dll		
204E34 NumberOfFunc 2 204E38 NumberOfNames 2				
204E38 NumberOfNames 2				
204E3C AddressOfFunc 204E	E48			
204E40 AddressOfNames 204E	E50			
204E44 AddressOfNam 204E	E58			
Exported Functions [2 entries]				
Offset Ordinal Fun	nction RVA Na	ime RVA	Name	Forwarder
204E48 1 5EA(A0 204	4E6A I	ternalJob	
204E4C 2 AB50	50 204	4E75 S	SetPath	

BumbleBee internal name, export functions, and TimeDateStamp

In the image below, we found the BumbleBee internal name and export function inside the process Rundll32.exe that executed the BumbleBee DLL loader:

-																																	
rundll32	.exe	(157	6) (0	x29	4872	2240	- 00	0x29	9487	26b	000)																						
00000be0	00	00	00	00	ff	ff	ff	ff	00	00	00	00	40	00	00	00	f0	24	03	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000c00	00	00	00	00	50	62	22	00	00	00	00	00	ff	ff	ff	ff	00	00	00	00	18	00	00	00	80	d5	00	00	00	00	00	00	Pb"
00000c20	00	00	00	00	00	00	00	00	00	00	00	00	80	58	22	00	00	00	00	00	ff	ff	ff	ff	00	00	00	00	40	00	00	00	X"@
00000c40	fO	lc	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	90	4a	22	00	00	00	00	00	ff	ff	ff	ff	J"
00000c60	00	00	00	00	40	00	00	00	00	b0	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	20	63	22	00	@
00000c80	00	00	00	00	ff	ff	ff	ff	00	00	00	00	40	00	00	00	d0	ad	01	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000ca0	03	00	00	00	60	4b	20	00	d0	49	20	00	00	4c	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	·····K ··I ··F ······
00000cc0	00	00	00	00	18	66	22	00	00	00	00	00	11	II	11	II	00	00	00	00	28	00	00	00	a8	86	03	00	00	00	00	00	f"
00000ce0	00	00	00	00	00	00	00	00	00	00	00	00	00	α6 4 -	00	00	00	00	00	00	98	4a	20	00	00	00	00	00	00	00	00	00	M
0000000000	00	00	00	00	00	00	00	00	-02	00	20	00	70	4a	20	00	00	40	20	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000420	00	00	00	00	10	40	20	00	00	40	20	00	00	4a	20	00	00	49	20	00	da	40	20	00	00	00	00	00	00	44	20	00	······ ··· ··· ··· ··· ··· ··· ··· ···
000000460	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	90	67	22	00	00	00	00	00	00 FF	40 66	20 FF	- 00 - FF	a"
00000480	00	00	00	00	18	00	00	00	84	84	03	00	00	00	00	00	00	00	00	00	00	00	00	00	02	00	00	00	40	10	20	00	т
00000da0	00	40	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	d8	68	22	00	00	00	00	00	ff	ff.	ff	ff	T. h"
00000dc0	00	00	00	00	28	00	00	00	e8	86	03	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	04	84	03	00	
00000de0	00	00	00	00	20	4d	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	38	65	22	00	
00000e00	00	00	00	00	ff	ff	ff	ff	00	00	00	00	28	00	00	00	68	86	03	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000e20	00	00	00	00	8c	96	4c	62	00	00	00	00	5c	4e	20	00	01	00	00	00	02	00	00	00	02	00	00	00	48	4e	20	00	LbN
00000e40	50	4e	20	00	58	4e	20	00	a0	5e	00	00	50	ab	00	00	6a	4e	20	00	75	4e	20	00	00	00	01	00	4c	64	72	41	PN .XN^PjN .uNIdrA
00000e60	64	64	78	36	34	2e	64	6c	6c	00	49	74	65	72	6e	61	6c	4a	6f	62	00	53	65	74	50	61	74		00	00	00	00	ddx64.dll.IternalJob.SetFath
00000e80	70	50	20	00	00	00	00	00	00	00	00	00	da	5d	20	00	00	41	16	00	00	56	20	00	00	00	00	00	00	00	00	00	pF]AV
00000ea0	46	5e	20	00	90	46	16	00	70	4f	20	00	00	00	00	00	00	00	00	00	e8	5f	20	00	00	40	16	00	b8	55	20	00	F^FpO@U .
00000ec0	00	00	00	00	00	00	00	00	2a	60	20	00	48	46	16	00	c 0	56	20	00	00	00	00	00	00	00	00	00	9a	60	20	00	*` .HFV` .
00000ee0	50	47	16	00	68	55	20	00	00	00	00	00	00	00	00	00	a4	60	20	00	f8	45	16	00	58	55	20	00	00	00	00	00	PGhU`EXU
00000f00	00	00	00	00	са	60	20	00	e8	45	16	00	60	50	20	00	00	00	00	00	00	00	00	00	e4	60	20	00	fO	40	16	00	`E`P`@
00000f20	30	56	20	00	00	00	00	00	00	00	00	00	32	61	20	00	c0	46	16	00	20	50	20	00	00	00	00	00	00	00	00	00	CV2aF F
00000f40	00	62	20	00	b0	40	16	00	d0	55	20	00	00	00	00	00	00	00	00	00	4a	62	20	00	60	46	16	00	00	00	00	00	.b@UJb .`F
00000f60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	62	5e	20	00	00	00	00	00	7a	5e	20	00	00	00	00	00	b^z^
00000f80	92	5e	20	00	00	00	00	00	aa	5e	20	00	00	00	00	00	ba	5e	20	00	00	00	00	00	d2	5e	20	00	00	00	00	00	·^ ·····^ ·····^ ·····
00000fa0	e8	5e	20	00	00	00	00	00	da	5f	20	00	00	00	00	00	са	5f	20	00	00	00	00	00	b6	5f	20	00	00	00	00	00	·^ ····· ····· ····· ·····
00000100	a4	51	20	00	00	00	00	00	8e	51	20	00	00	00	00	00	7c	51	20	00	00	00	00	00	68	51	20	00	00	00	00	00	<u></u>
00000fe0	56	51	20	00	00	00	00	00	46	51	20	00	00	00	00	00	34	5f	20	00	00	00	00	00	22	51	20	00	00	00	00	00	V <u>4_</u> <u>*_</u>
00001000	ue co	51	20	00	00	00	00	00	Ia	se	20	00	00	00	00	00	52	se Cl	20	00	00	00	00	00	00	00	00	00	00	00	00	00	·
00001020	00	61	20	00	00	00	00	00	/e	61	20	00	00	00	00	00	90	61	20	00	00	00	00	00	9a	00	20	00	00	00	00	00	a~aaa
00001040	ac	01	20	00	00	00	00	00	40	01	20	00	00	00	00	00	se	10	20	00	00	00	00	00	00	00	00	00	00	00		00	.aNa>a

Bumblebee's internal name and the export functions names in the memory By inspecting the unpacked BumbleBee sections, we discovered that the .data section contains two executables:

property	value	value	value	value	value	value	value
name	.text	.rdata	.data	.pdata	.gfids	.tls	.reloc
md5	7855722A7B96091D08D348E	66C85B9E435112AAEDD1C1	07A36690C4A70C0EE985D806FF1F25AD	490087E1686172414BBE5439	1429057928C52BB91F168CF	BE966AE9956FBCDBEB8EDF	FD7C575D75D1F69FC4C79D
entropy	6.409	5.667	5.264	6.194	3.579	0.003	5.440
file-ratio (98.23%)	60.35 %	27.68 %	5.39 %	3.56 %	0.02 %	0.21 %	1.02 %
raw-address	0x00001000	0x00164000	0x00207000	0x0022D000	0x00242000	0x00243000	0x00245000
raw-size (2361856 bytes)	0x00162400 (1451008 bytes)	0x000A2800 (665600 bytes)	0x0001FA00 (129536 bytes)	0x00014E00 (85504 bytes)	0x00000200 (512 bytes)	0x00001400 (5120 bytes)	0x00006000 (24576 bytes)
virtual-address	0x000000080001000	0x000000080164000	0x000000080207000	0x00000008022D000	0x000000080242000	0x000000080243000	0x000000080245000
virtual-size (2385799 bytes)	0x0016234E (1450830 bytes)	0x000A270C (665356 bytes)	0x00025B1C (154396 bytes)	0x00014DF0 (85488 bytes)	0x000001D4 (468 bytes)	0x000013D1 (5073 bytes)	0x00005E7C (24188 bytes)
entry-point	0x0012BC18						
characteristics	0x60000020	0x40000040	0xC0000040	0x40000040	0x40000040	0xC0000040	0x42000040
writable	•	÷	x			x	
executable	x						
shareable							
discardable							х
initialized-data		х	x	х	х	х	х
uninitialized-data							
unreadable							
self-modifying							
virtualized							
file		-	executable, offset: 0x00214720, size: 27492	-	-		-
file			executable, offset: 0x0021B520, size: 25460				

PEStudio shows the unpacked Bumblebee section and highlighted the .data section

We extracted the two hidden payloads from the .data section by using Hex-Editor tool:

0021450 00
00214650 00 50 1A 16 00 00 00 16
00214660 00
00214620 00 C0 00 14 0 00 00 00 00 00 00 00 00 00 00 00 00
00214680 00
00214650 70 C4 1D 87 94 02 00 00 01 00 00 00 00 00 00 0, A. +*
00214620 88 C4 1D 87 94 02 00 00 C2 00 00 00 00 00 00 0 Å .**
00214620 A0 C4 1D 87 94 02 00 00 77 00 00 00 00 00 00 A, 4*"
002146C0 B8 C4 D D 00
00214620 00 00 00 00 00 00 00 00 00 00 00 00 0
00214F82D 050 000
00214700 00 00 00 00 00 00 00 00 00 00 00 00 00
00214710 00
Collign Excent (text) 00214720 FL 53 90 00
00214730 B8 00 00
00214740 00
00214750 00 00 00 00 00 00 00 00 00 00 00 00 0
00214760 0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68, f, i, , Lî ITh 00214770 72 0F 77 26 16 DD 20 63 61 6E 6E 6F is program canno 00214760 74 20 62 20 72 75 67 20 16 0 F7 20 66 6F 20 44 4F 63 20 + be win in DNS Checksum Search (13 hits) Offset Excent (hex) 0 4 D 5A 90 00 30 00 00 00 00 00 00 00 00 00 00 00
00214770 69 73 20 60 67 72 61 62 72 75 20 60 65 72 75 20 60 65 72 75 20 60 65 72 75 20 60 65 72 75 20 60 65 72 75
Born 214280 74 20 F2 20 64 44 45 23 20 F F0 F0 Offset Excerpt (hex) Excerpt (hex) Excerpt (hex) Excerpt (hex) Excerpt (hex) MZ <
Search (13 hits) Excert (14 hits) Offset Excert (18 hits) Excert (18 hits) 0 4D 5 A 90 00 30 00 00 00 00 00 00 00 00 00 00 00
Fr Offset Excertol (hex) Excertol (hex) 0 4D 5A 90 00 03 00 00 00 40 00 00 0FF FF 00 00 B8 00 00 00 00 00 00 00 00 00 00 00 00 00
O 40 5A 00 00 30 00 00 04 00 00 00 FF FF 00 00 BB 00 00 00 00 00 00 00 00 00 00
0 4D 5A 90:00 50:00 00 00 00 00 00 00 00 00 00 00 00 00
3BPC/ 194 5 33 0B 48 01 FA FF F0 00 43 0F B7 C3 85 4D 3A 00 04 4C 86 C4 60 46 40 F6 22 46 0B 80 10 00 00 48 86 L300 H04/C8F C4 60 80 10 00 00 42 0B C4 26 40 F6 70 38 0C 46 04 60 00 04 48 86 12B410 20 0F 00 00 CC CC CC CC CC 48 83 EC 18 4C 86 C1 88 4D 5A 00 00 46 58 C2 46 0B 80 10 00 00 48 86
SAAFF 48 08 D7 E8 39 36 02 00 88 53 24 40 39 38 CE E8 4D 5A 00 00 48 88 5C 47 60 08 00 100 00 04 88 8 FraceFox.15gR(4eMZ,Fr45),H45 128410 200 F0 00 CC CC CC CC 48 88 EC 148 42 86 18 84 05 A0 00 06 63 90 87 53 34 86 34 82 C4 80 38 C Imilini,FL1,AMZ,F9A,K9),uHr. 1326E4 75 4A 33 C9 FF 15 4A 1A 03 00 48 85 C0 74 3D B9 4D 5A 00 00 66 39 08 75 33 48 63 48 3C 48 03 C8 Imilini,FL1,AMZ,F9A,K9),uHr. 159304 5C 24 30 48 83 C4 20 5 FC 3C CC CC 48 88 C1 B9 4D 5A 00 00 66 39 08 77 03 33 C0 C3 48 63 48 3C S0H/A Ž, ŽIIIH-LÁ'NZ,F9A,SJÄH-L+ 19DFBD 70 00 B4 E1 34 D3 FB 59 E8 88 AB 57 27 49 04 66 4D 5A F5 03 88 BA 09 52 D C0 D3 22 D C8 07 42 93 A1 17 E1 F p. Sá4Ó Vec «W1 (FMZ 6, "9, QI'4), 214720 00 00 00 00 00 00 00 00 00 00 00 00 00
128410 20 0F 00 00 C C C C C C C C C C 48 35 C 18 4 C 86 C 188 4 D 5 A 00 00 65 90 0E 5 34 86 54 90 C F 15 94 86 30 5 IIIII+/LL-A,ML.:F9.4K8/vjv;HC. 13265 L - T, 54 A 33 C 9F F1 54 A 1 A 03 04 88 5C 07 4 30 B 94 D 5 A 00 00 66 39 08 73 33 48 63 48 3C IIIII+/LL-A,ML.:F9.4K8/vjv;HC. 159304 5C 24 30 48 83 C 4 20 5F C3 CC C C C 48 8B C1 B9 4D 5A 00 00 66 39 08 74 03 33 CO C3 48 63 48 3C \S0H fÅAIIIH+(A ⁺ MZ:F9.4X3ÅH;CH<
1326E4 75 4A 33 C 9FF 15 4A 1A 03 00 48 85 C 07 43 00 94 05 5A 00 00 66 39 08 75 33 48 03 48 32 C 49 03 C 8 U3EyJHAt= MZ.HS.JMCH <h.e< td=""> 159304 5C 24 30 48 83 C 42 05 F C 3 C C C C C 48 8B C 1 89 4D 5A 00 00 66 39 08 74 03 33 C 0C 34 86 348 3C V50HJÄ ÄÄIIH-Å MZ.J.(F9.J3ÅHCH<h.e< td=""> 1997B0 70 00 B4 13 40 3F B5 9E B8 8B A5 72 49 04 66 4D 5A F 50 38 8B A0 05 2D C B0 34 29 3A 11 7E 1F p. / 4d/0 åVte-«W1 (MZč., *%) (*%) (*%)</h.e<></h.e<>
159304 5 C 24 30 48 33 C 42 05 FC 35 C C C C C 48 08 C 1 89 4D 5A 00 00 66 59 08 74 03 32 C C 34 86 34 93 C \S0H7A_AIIIH-A'MZ_59:t3AAH-CH 19DF8D 70 00 B4 E1 34 D3 FB 59 E8 88 AB 57 27 49 04 66 4D 5A F5 03 88 BA 00 52 D C B0 34 29 3A 11 7E 1F p: 54Ó0Yec«VI; MZ8.°9.0°4);, 1DDCEE E2 59 3F A0 14 C 4C CA 21 75 A4 FC 18 P5 A7 88 4D 5A 32 D 10 30 F9 48 77 82 5A 7E 24 7C 37 18 15 p: 54Ó0Yec«VI; MZ8.°9.0°4);, 214720 00 00 00 00 00 00 00 00 00 00 00 00 00
190FB0 70 00 84 E1 34 03 FB 59 EB 88 AB 57 27 49 04 66 4D 5A F 50 38 BA 00 52 0 C 80 34 29 3A 11 7E 1F p.'340'0We-sW1'HMZ^5-'%D'4), 1D0CEE E2 59 3F A0 14 C 4E CA 21 75 A4 F C8 19 F5 A7 88 4D 5A 32 1D 30 F9 48 77 82 5A 7E 24 7C 37 1B 15 àY? Ait<20E.654 MZ2.00Hw,Z-5]7
IDDCCEE E2 59 3F A0 14 C4 EC A2 17 5A 4F C8 19 F5 A7 88 4D 5A 32 10 30 F9 48 77 82 5A 7E 24 7C 37 18 15 áY?.Åic.ZOE.ôs·MZ2.00Hw,Z~s[7 214720 00 00 00 00 00 00 00 00 00 00 00 00 00
214720 00 00 00 00 00 00 00 00 00 00 00 00 00
217057 0 60 00 08 84 55 84 89 45 F4 40 F 87 00 30 4D 5A 00 00 88 D2 D2 60 A9 89 F5 29 D CE 0F 44 €%seEx.ex=MZ6V%c ⁵ R.I.D 218520 00 00 00 00 00 00 00 00 00 00 00 00 00
218520 00 00 00 00 00 00 00 00 00 00 00 00 00
21E0F0 24 20 48 89 44 24 78 48 8B 44 24 78 0F B7 00 3D 4D 5A 00 00 74 07 33 C0 E9 B7 03 00 00 48 8B 44 \$H%oD\$x+-=MZ.t.t3AéH <d< td=""></d<>
2389BC A0 59 0C 00 C2 59 0C 00 BC 6B IF 00 D0 59 0C 00 4D 5A 0C 00 BC 6B IF 00 50 5A 0C 00 81 5A 0C 00 YÅY¼kĐY MZ ¼kĐY MZ ¼kĐY

Hex-Editor shows 3 MZ headers: the first one is the Bumblebee, and the other two are additional payloads

The first payload from the .data section is a 32-bit DLL payload:

property	value
md5	36D49170F3115D378F8B6A3A45B23525
sha1	AE1A95DA9B7488B51C8549C52DE8E2F73C022608
sha256	EED2D5DD3B0FCCD71FA30B79708004E7393E83AAC8566E80808F1162936BC1F2
md5-without-overlay	n/a
sha1-without-overlay	n/a
sha256-without-overlay	n/a
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 40 00 00 00 00 00
first-bytes-text	M Z @
file-size	28160 (bytes)
size-without-overlay	n/a
entropy	6.323
imphash	1369F81AACB871DA7C04248B77211BB2
signature	n/a
entry-point	55 8B EC 51 8B 45 0C 89 45 FC 83 7D FC 01 74 02 EB 0D 8B 4D 08 51 8B 55 10 52 E8 C1 FF FF B8 01
file-version	n/a
description	n/a
file-type	dynamic-link-library
сри	32-bit
subsystem	GUI
compiler-stamp	0x624C9623 (Tue Apr 05 12:18:59 2022)
debugger-stamp	0x624C9623 (Tue Apr 05 12:18:59 2022)
resources-stamp	n/a
import-stamp	0x00000000 (empty)
exports-stamp	0x624C9623 (Tue Apr 05 12:18:59 2022)
version-stamp	n/a
certificate-stamp	n/a

PEStudio showing the payload's metadata

We found a few interesting functions in the payload strings indicating that this payload has process injection capabilities. For example, "CreateProcess," "NtWriteVirtualMemory," "CreateRemoteThread," and "WinExec."

encoding (2)	size (bytes)	location	blacklist (82)	hint (54)	value (382)
ascii	19	0x00005ED8	x	-	<u>NtReadVirtualMemory</u>
ascii	19	0x00005EEC	x	1	NtFreeVirtualMemory
ascii	23	0x00005F00	x		<u>NtAllocateVirtualMemory</u>
ascii	14	0x00005F18	x		NtResumeThread
ascii	18	0x00005F28	x		NtSetContextThread
ascii	23	0x00005F3C	x	20	<u>NtSetInformationProcess</u>
ascii	22	0x00005F54	x	-	NtSetInformationThread
ascii	15	0x00005F6C	x		NtSuspendThread
ascii	20	0x00005F7C	x	-	<u>NtUnmapViewOfSection</u>
ascii	11	0x00005FC8	x	100	NtOpenEvent
ascii	20	0x00005FF4	x	-	<u>NtWriteVirtualMemory</u>
ascii	25	0x0000600C	x	e	<u>NtQueryInformationProcess</u>
ascii	23	0x00006028	x		<u>NtAdjustPrivilegesToken</u>
ascii	18	0x0000605C	x	123	<u>NtTerminateProcess</u>
ascii	13	0x00006070	x	10	<u>NtOpenProcess</u>
ascii	13	0x00006080	x	(46)	NtOpenSection
ascii	17	0x000060B4	x	G	<u>RtlExitUserThread</u>
ascii	19	0x000060C8	x	÷	<u>KiUserApcDispatcher</u>
ascii	25	0x000060DC	x	-	<u>KiUserExceptionDispatcher</u>
ascii	12	0x000060F8	x	1	<u>NtOpenThread</u>
ascii	19	0x00006108	x	(H)	RtIDecompressBuffer
ascii	13	0x000061A0	x	172	CreateProcess
ascii	21	0x000061B0	x	e.	CreateProcessInternal
ascii	21	0x000061C8	x	140	CreateProcessInternal
ascii	13	0x000061E0	x	178	CreateProcess
ascii	18	0x000061F0	x		CreateRemoteThread
ascii	15	0x00006204	x		FindFirstFileEx
ascii	15	0x00006218	x	174	FindFirstFileEx
ascii	31	0x00006288	x	-	<u>RtlInstallFunctionTableCallback</u>
ascii	7	0x000062A8	x	(4) (4)	<u>WinExec</u>
ascii	18	0x00006310	x	172	CreateRemoteThread
ascii	13	0x0000634C	x	40	FindFirstFile
ascii	13	0x0000635C	x	545 ·····	FindFirstFile
ascii	16	0x000065AA	x		PathFindFileName

PEStudio showing the payload's strings that could be related to process injection

The second payload that we extracted from the .data section is a 64-bit DLL payload:

property	value
md5	FC3535258586EAF20511A45BA099D14E
sha1	425E0123D2BE8B84F7B58B89A5B06711EA564344
sha256	FAC5701CCAC0C1AC224FD601EFD6DBBCF867E0BFA02AFE336ADAFA80E0399A45
md5-without-overlay	7476D0AA4BA606A51450093ECF0086ED
sha1-without-overlay	F181B15536CE3D28B8603972000B8192F5B3D04C
sha256-without-overlay	E1B7382F6D5588DED0BB9BC305CC9CF2D11272DFB0CCC0584F915D7B9B48746B
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 0F FF 00 00 B8 00 00 00 00 00 00 00 40 00 00 00 00 00
first-bytes-text	M Z
file-size	195296 (bytes)
size-without-overlay	26112 (bytes)
entropy	5.480
imphash	20A787DCB5EC1605108FA6BA85DA6A52
signature	n/a
entry-point	4C 89 44 24 18 89 54 24 10 48 89 4C 24 08 48 83 EC 38 8B 44 24 48 89 44 24 20 83 7C 24 20 01 74 02
file-version	n/a
description	n/a
file-type	dynamic-link-library
сри	64-bit
subsystem	GUI
compiler-stamp	0x624C962D (Tue Apr 05 12:19:09 2022)
debugger-stamp	0x624C962D (Tue Apr 05 12:19:09 2022)
resources-stamp	0x00000000 (empty)
import-stamp	0x00000000 (empty)
exports-stamp	0x624C962D (Tue Apr 05 12:19:09 2022)
version-stamp	n/a
certificate-stamp	n/a

PEStudio showing the payload's metadata

We analyzed the payload binary and noticed that this payload is responsible for communicating with BumbleBee's C2 server:

DOS Header	Offset	Strings recognized ASCII
	00008680	45 147 229 177-443
V NT Header	0000000	45.14ALED.TATATO
🗸 🔢 Optional Header		
Data Directories		
Section Headers		
DIRECTORY_ENTRY_EXPORT		
DIRECTORY_ENTRY_IMPORT		
DIRECTORY_ENTRY_EXCEPTION		
2 DIRECTORY_ENTRY_BASERELOC		
MIRECTORY_ENTRY_DEBUG		
AppManifest		

In the strings we can see the C2 server's IP address and port

Both DLL payloads have the same internal name "RapportGP.dll." An interesting point regarding the payloads internal name is that there is a legitimate DLL named "RapportGP.dll" that is part of a "Trusteer Ltd" product from a computer security division of IBM.

File Settings View Compare Info								File Settings View Compare	Info							
 ▼ P2rundl32.excbin > D05 Header > D05 Header > D05 Header > D05 Header > Strong Header > Section Headers > Secti	 AA60 € AA60 € AA70 € AA80 €<td>40 40 40 40 1 2 3 4 5 0 1 2 3 4 5 0 1 2 3 4 5 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 1 0<td>x y x y x x y x y y x x y x y y x x y y x y y x x y</td><td>B C D E F C1 F F T S C1 F F T S C1 C C C T T C1 C C C C T T C1 C C C C T<td>0 1 2 9 4 9 0 1 2 9 0 1 1 2 9 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</td><td>6 7 8 9 A B C 1</td><td></td><td>(in Carany Unit Costing</td><td></td><td>4A60 4A70 4A90 4A80 4A80 4A80 4A80 4A80 4A80 4A80 4A8</td><td>0 1 3 4 0 1 2 3 4 0 1 2 3 4 0 5 8 5 8 5 0 1 0 0 0 0 5 0 1 0 0 0 0 0 5 10 0 0 0 0 0 0 0 0 3 3 7 1 0 0 0 0 0 0 3 4 0 5 0 5 0</td><td>** * 6 7 8 9 00 85 84 90 85 10 85 10 80 85 10 80 90 74 00 00 10 70 70 10 10 0 6024C9623 00 74 00 10 7DE8 1 0</td><td>A B C D Z F A B C T F 0.1 F T B C T T B C T T B C T T B C C C D S T T T B C C C D S T D C C D S D D T T D D T D D T D D T D</td><td>0 1 2 3 4 0 1 2 3 4 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2</td><td>S (7 (9) A B) E . E 0</td><td></td></td></td>	40 40 40 40 1 2 3 4 5 0 1 2 3 4 5 0 1 2 3 4 5 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 1 0 <td>x y x y x x y x y y x x y x y y x x y y x y y x x y</td> <td>B C D E F C1 F F T S C1 F F T S C1 C C C T T C1 C C C C T T C1 C C C C T<td>0 1 2 9 4 9 0 1 2 9 0 1 1 2 9 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</td><td>6 7 8 9 A B C 1</td><td></td><td>(in Carany Unit Costing</td><td></td><td>4A60 4A70 4A90 4A80 4A80 4A80 4A80 4A80 4A80 4A80 4A8</td><td>0 1 3 4 0 1 2 3 4 0 1 2 3 4 0 5 8 5 8 5 0 1 0 0 0 0 5 0 1 0 0 0 0 0 5 10 0 0 0 0 0 0 0 0 3 3 7 1 0 0 0 0 0 0 3 4 0 5 0 5 0</td><td>** * 6 7 8 9 00 85 84 90 85 10 85 10 80 85 10 80 90 74 00 00 10 70 70 10 10 0 6024C9623 00 74 00 10 7DE8 1 0</td><td>A B C D Z F A B C T F 0.1 F T B C T T B C T T B C T T B C C C D S T T T B C C C D S T D C C D S D D T T D D T D D T D D T D</td><td>0 1 2 3 4 0 1 2 3 4 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2</td><td>S (7 (9) A B) E . E 0</td><td></td></td>	x y x y x x y x y y x x y x y y x x y y x y y x x y	B C D E F C1 F F T S C1 F F T S C1 C C C T T C1 C C C C T T C1 C C C C T <td>0 1 2 9 4 9 0 1 2 9 0 1 1 2 9 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</td> <td>6 7 8 9 A B C 1</td> <td></td> <td>(in Carany Unit Costing</td> <td></td> <td>4A60 4A70 4A90 4A80 4A80 4A80 4A80 4A80 4A80 4A80 4A8</td> <td>0 1 3 4 0 1 2 3 4 0 1 2 3 4 0 5 8 5 8 5 0 1 0 0 0 0 5 0 1 0 0 0 0 0 5 10 0 0 0 0 0 0 0 0 3 3 7 1 0 0 0 0 0 0 3 4 0 5 0 5 0</td> <td>** * 6 7 8 9 00 85 84 90 85 10 85 10 80 85 10 80 90 74 00 00 10 70 70 10 10 0 6024C9623 00 74 00 10 7DE8 1 0</td> <td>A B C D Z F A B C T F 0.1 F T B C T T B C T T B C T T B C C C D S T T T B C C C D S T D C C D S D D T T D D T D D T D D T D</td> <td>0 1 2 3 4 0 1 2 3 4 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2</td> <td>S (7 (9) A B) E . E 0</td> <td></td>	0 1 2 9 4 9 0 1 2 9 0 1 1 2 9 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	6 7 8 9 A B C 1		(in Carany Unit Costing		4A60 4A70 4A90 4A80 4A80 4A80 4A80 4A80 4A80 4A80 4A8	0 1 3 4 0 1 2 3 4 0 1 2 3 4 0 5 8 5 8 5 0 1 0 0 0 0 5 0 1 0 0 0 0 0 5 10 0 0 0 0 0 0 0 0 3 3 7 1 0 0 0 0 0 0 3 4 0 5 0 5 0	** * 6 7 8 9 00 85 84 90 85 10 85 10 80 85 10 80 90 74 00 00 10 70 70 10 10 0 6024C9623 00 74 00 10 7DE8 1 0	A B C D Z F A B C T F 0.1 F T B C T T B C T T B C T T B C C C D S T T T B C C C D S T D C C D S D D T T D D T D D T D D T D	0 1 2 3 4 0 1 2 3 4 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2	S (7 (9) A B) E . E 0	
	63E4	AddressOfNam	0							6364	AddressOfNam	0				
	Exported Fi Offset	Unctions [0 entries]	Function RVA	Name RVA	Name	Forwarder				Exported	d Functions [0 entries] Ordinal	Function RVA	Name RVA	Name	Forwarder	
and the second se									n] rundi132.exc.bin		ound				, on a de	
Loaded: C:/Users/user/Desktop/P2 rundll32.exe.bir	n					Check	for updates								CP CP	neck for updates

Payloads internal name and TimeDateStamp

Final notes

BumbleBee threat actors are not the first to change the initial access method from malicious office documents to malicious ISO image files. The ISO image file abuse was also seen a few years ago, but in recent months, we have observed an increase in "ISO campaigns."

Different threat actors abuse ISO image files to deliver their payloads. For example, BazarISO deploys Bazarloader, and IcedID started to use ISO image files instead of MalDocs like in the two examples below.

💿 l 🔽 📗 Ŧ	I		Manage	Manage	E:\		
File Hom	e Share	View	Shortcut Tools	Application Tools			
$\leftrightarrow \rightarrow \cdot$	> This	PC → DVI	D Drive (E:)				
> 🔮 Docur	ments	^	Name	^	Date modified	Туре	Size
> 🕹 Down	loads		😤 autorun.exe		1/21/2022 6:15 AM	Application	673 KB
> ☆ Favori	tes		🔒 docs		1/21/2022 6:15 AM	Shortcut	2 KB
🛃 Links							
> 🔒 Local	Settings						
> 📙 Micro	softEdgeBacku	ps					
> 👸 My Do	ocuments						

Documents-17.iso (Bazarloader)



Invoice_pdf_1.iso (IcedID)

In most of the cases, we've seen that during different IR cases, the campaigns escalated to full-blown ransomware attacks. We believe that IAB groups work and collaborate with ransomware affiliates like CONTI, LockBit, AvosLocker, and more. For example, we observed an IcedID infection that leads to CONTI ransomware attack (<u>Shelob Moonlight</u>)...

The Orion team is constantly monitoring BumbleBee and the IAB group's activities closely and analyzing them to better understand their motivation. As we learn more, we will publish our findings and artifacts to share additional insights for BumbleBee infection to ransomware post-attack chain.

We're expecting to see more malware campaigns that will use the ISO delivery method in the near future. So, stay vigilant.

As a final note, we'd like to share these indicators of compromise with you.

Indicators of compromise:

BumbleBee payload

88F5AE9691E6BCDD4065A420EAFAF3E3AA32C69605BF564A42FFD8ECD25C9920 4a49e2f06ba48d3a88fdeb83fb8021f3d165535e8ea5319b16a7ebe4da9c0751 08cd6983f183ef65eabd073c01f137a913282504e2502ac34a1be3e599ac386b 186145f84ed6a473ec6bc4afa66bff156057888938793b12afd17659041ddbba 4063fab9176db3960fa6014173b6c7ba52f19424887f5a6205ff73aa447ada61 53b3ebaa3c485772f8e6abaa0f366ef192137496a7064e015ced4e6fc204b3c8 d74a3f9b35d657516eb53d4e70582f93d22077d3e0936758cc4ef76d5171075d 8f47c3962a7c418bae71fec42bbca9524b72f8f0fd2dd81d1175138f7d20b2f7 c97b8bffcbe424cbc2a6e1135068d071c6f4e8f020fccd2db3dbee3aa80102ac

BumbleBee C2 server

IP: 23.82.19[.]208 Port 443
IP: 192.236.198[.]63 Port 433
IP: 45.147.229[.]177 Port 433
Cobalt Strike C2 server
hojimizeg[.]com - 45.147.228[.]197
notixow[.]com - 23.19.58[.]154
rewujisaf[.]com - 142.234.157[.]176

We hope this was helpful. And remember to check our blog page and follow us on social media to see when we publish updates.

Have questions? Let us know.